Examining the Aufbau Principle and Ionization Energies: A Computational Chemistry Exercise for the Introductory Level

Irene K. Metz, Joseph W. Bennett, and Sara E. Mason*

Read Online

ACCESS | Metrics & More | Article Recommendations | SI Supporting Information

ABSTRACT: While computational chemistry continues to play a growing role in chemical research, issues with access and the timing of when these tools are introduced to chemistry students remain as barriers to wider interest and use. The accessibility of free software for chemical modeling has increased in recent years, promoting the inclusion of computational exercises at small undergraduate or secondary education institutions. However, as many computational chemistry methods are based on advanced theories, students are often not exposed to computational chemistry until organic or physical chemistry courses. This work looks at using an open-source code to introduce computational chemistry in first-year or introductory level courses while enhancing understanding of the Aufbau principle, electron configurations, and ionization energy. Additionally, this hands-on, interactive laboratory activity promotes important workplace skills such as collaboration, data analysis, graphical interpretation, and critical thinking.
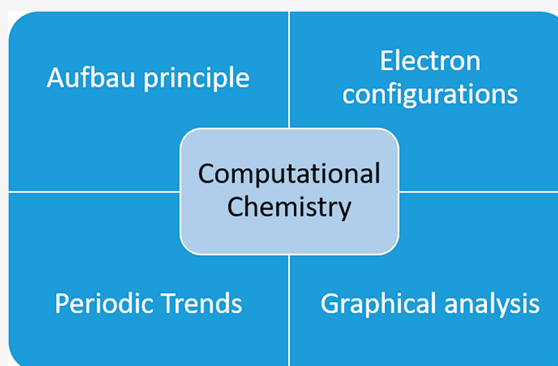
INTRODUCTION

Fundamental topics in quantum mechanics such as energy levels, wave functions, and electron configurations have commonly been taught through the lecture approach and are covered in the majority of introductory level chemistry courses. While these topics are important to create the foundations for students to understand periodicity and more advanced material, it is often found that students rely on rote memorization to learn the material[1] rather than truly understanding it. Additionally, when supplemental materials are used to address this topic, there are few interactive physical demonstrations that students can examine for themselves other than emission spectra that do not require expensive equipment. While simulations are available, they do not necessarily address the differences in energy levels, sublevels, wave functions, or electron configurations in a way that allows students to significantly interact beyond the clicking of mouse buttons. As a result, this remains an area that many introductory level students struggle with due to its abstract nature.[2,3]

Quantum mechanics ties directly to computational chemistry, which has been playing a growing role in research over the years. While computational tools are already in use at some institutions to aid in teaching of chemical bonding, VSEPR theory, and electron density,[4−6] to name a few, they largely have not been used to address the rules behind atomic structures. If we can use computational tools to allow students to derive data to explore the concepts of quantum mechanics, we provide students with a modern method of scientific inquiry where they can examine abstract topics through an interactive, hands-on approach, rather than relying on memorization. Additionally, by introducing students to computational tools early in their academic careers, we may help them learn more about applications in their desired fields of study.

Despite the benefits and potential to enhance learning, computational chemistry tools have not become more mainstream in the introductory and first-year chemistry curriculum. Grushow and Reeves suspect that this may be due to complications instructors encountered when they were first exposed to computational methods either as a student or early in their teaching careers.[7] The authors claim that beyond unfavorable past experiences, the sometimes high cost of licenses, equipment, maintenance, and updates as well as the long calculation run times for small molecules are obstacles that

4017

https://doi.org/10.1021/acs.jchemed.1c00700
J. Chem. Educ. 2021, 98, 4017−4025

many instructors have dealt with in years past. Therefore, while advances in hardware and code performance make it tractable to run calculations on small molecules during class periods, equipment and licensing costs, as well as software and hardware maintenance, are still significant concerns.[7] The availability of low-cost or free user-friendly software (Avogadro,[8] nwchem,[9] openchemistry,[10] jmol,[11] chemozart[12]) has led to a slight increase in the early introduction of computational chemistry to undergraduates, but the primary focus on the early introduction of computational chemistry appears to be on molecular modeling.[4−6,13−18]

What the scientific community knows about electron configurations has not changed in over half a century,[19−22] but the best practices for chemical education have evolved, and the cost and efficiency for computational chemistry have decreased significantly. Additionally, there are some literature references[21,22] that suggest computational methods could be used to rationalize electron configurations and help enhance student understanding. Previous authors creating laboratory activities focusing on electron configurations have noted that the standard exercises of examining line spectra do not assist in bridging the significant disconnect between students' lecture and laboratory experiences when discussing electronic structure.[3,23,24] Therefore, providing students with an interactive computational approach which allows them to engage with the material to explore the differences in energy levels, sublevels, and how to plot orbital energies could be beneficial. Through the use of software to collect previously inaccessible data, first-year students can explore the anomalies of electron configurations and be allowed to come to their own conclusions on why they occur. For students to get the most out of a computational experience, easy to follow examples must be provided to ensure those with minimal exposure to coding and computers can follow with minimal frustration.

In this paper, we present the details of an introductory chemistry laboratory activity using computational chemistry and designed to guide students to derive the Aufbau principle through a hands-on approach. This activity is designed to take place over one to two 2-h lab sessions to promote student cooperation, critical-thinking, quantitative reasoning, and a greater understanding of the Aufbau principle and its anomalies. The main goals of this exercise are that the students will be able to

- Identify and explain anomalies in electron configurations through data analysis.
- Create wave function plots and relate the nodes in the radial wave functions to relative energy levels.
- Build orbital energy diagrams, and explain the observed trends.
- Determine why the 4s electrons are listed prior to the 3d electrons, and explain electron removal trends.

Additionally, students will gain exposure to an open-source computational tool, allowing bridges to be created between previous coding experiences they may have been exposed to at their secondary institutions. The activity is assessed through questions incorporated into student laboratory reports, requesting that students identify and explain the correct electron configuration for elements based off presented data, determine electron removal trends, and create and interpret a graph of the 4s and 3d energies, explaining how it relates to the Aufbau principle.

ACTIVITY DESIGN

This 1−2 day laboratory exercise was designed to coincide with or follow lectures on introductory level electronic structure to enable students to derive answers to questions which often arise from a lecture only approach. This activity was tested over the course of five semesters (2018−2020) in 2 h long lab periods with between 3 and 7 sections of students per semester, which totals over 300 students. The students are those enrolled in either classes designed for health sciences, which is essentially the first semester of a general, organic, and biochemistry course sequence, or the standard first-year sequence of chemistry courses typically taught at both four-year institutions and community colleges. These courses consisted of a blend of commuting rural and urban students, many of whom have not been exposed to applications of computational chemistry. The number of lab periods this activity takes place over depends on the course, with the students in the health sciences chemistry course doing only the first day of the activity, while those in the standard first-year sequence course do both days. It is recommended that the students have received a lecture on quantum numbers and electron configurations prior to the activity as shown in Figure 1, and be familiar with graphing either in Excel or Google sheets.
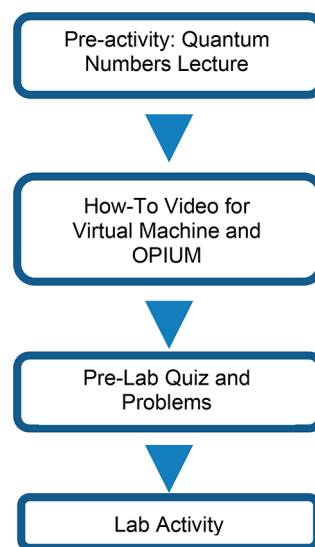
Figure 1. Flowchart of student process from lecture to lab activity for Aufbau computational exercise.

Following the completion of their portion of the activity, students turn in data for their assigned elements to a class database, and write up a laboratory report, creating and including graphs as necessary to answer the assigned data analysis questions. If questions were missed or skipped, students were contacted to discuss the data and graphs, and encouraged to correct and resubmit their reports. The evaluation of the final reports and additional exam questions allowed the achievement of the learning objectives to be assessed. Descriptions of the activities carried out on each day of the laboratory experiment are described below, and the complete instructor and student guides can be found in the Supporting Information. Student exercises can also be found at the Mason group web page, QCC for Everyone.[25]

Software

For the goal at hand of having introductory chemistry students identify correct electron configurations and create graphs ranking the relative energies of different electron configurations for a given atom, a user-friendly code capable of calculating all-electron energies is desirable. The all-electron energies and wave functions produced by the code are a result of the quantum mechanical calculations carried out as preliminary steps prior to the use of a pseudopotential. The code needs to be low-cost to be implemented at the community college level and self-training resources must be available. The Open-source Pseudopotential Interface/Unification Module (OPIUM)[26] is a Linux-driven, open-source software package designed for the generation of pseudopotentials to use within density functional theory calculations that can be utilized to investigate atomic, molecular, bulk, and surface properties of materials. When paired with the Grace[27] plotting software, it is capable of producing plots of radial wave functions computed based on quantum mechanics, which can provide visualization of periodic trends in atomic radii. A computer running a Linux operating system is not required, as "virtual machines", which emulate a Linux environment, can be used. This allows one to run the Linux operating system in an app window, mimicking a separate computer, providing uniform access to the code, effectively minimizing technological barriers to completing the lessons. Due to the short calculation times of OPIUM, it can be run entirely in the virtual environment, without affecting other processes or operating systems of the parent machine. It is important to note that our use of OPIUM,[26] to compute the total energies of atoms/cations with varying valence electron configurations, is not the primary function of the code, which is pseudopotential generation. Therefore, certain functions of the code are not accessed in the exercises presented in the Supporting Information.

Both Grace[27] and OPIUM[26] can be installed in any virtual Linux machine within a Windows-operating system. This allows the code to be used on pre-existing equipment, rather than purchasing computers possessing solely Linux operating system. For this experiment, the binary -executables of OPIUM[26] and Grace[27] are installed on the virtual machine, and one can then use lessons and the up-to-date tutorials available on the Mason group web page, QCC for Everyone .[25] Once the virtual machine and any desired example files are set up to instructor preference, the virtual machine can then be copied to any machine within the network by the college's IT department.

Preliminary Activities

Prior to starting the activity, students attend a lecture on the basics of electronic structure, including an introduction to *nlm* quantum numbers, nodes, orbitals, radial wave functions, and electron configurations, in order to familiarize them with the necessary topics for the activity. Students are then assigned two to three elements from the periodic table for the activity. The assigned elements typically range from nitrogen to barium, as early elements do not possess many orbitals, and each student is assigned an element within either the 3 or 4*d*-block. Elements after barium possess *f* orbitals, and require a greater level of complexity for the input file, which is beyond the scope of introductory level courses. A 15 min how-to video[25] is made available to the students prior to the lab in order to familiarize them with the virtual machine, OPIUM[26] software and files, Grace,[27] and the Linux commands necessary to operate the software. Students are provided with an accompanying Power-

Point presentation to encourage them to take notes and are able to pause and rewatch the video as necessary. A preactivity quiz follows, to ensure students know the components required for the input files, how to access the output files, and the necessary commands. Students are also required to turn in copies of their three desired test configurations for each of their assigned elements prior to the start of the activity. One of the configurations must be the correct or literature electron configuration, but the others can be any variation the student wishes to test, as long as the total number of electrons does not change for the neutral configurations.

Description of the Activity

OPIUM[26] is well-suited for use in teaching the Aufbau principle and for determining ionization energies. The code takes seconds to calculate the all-electron eigenvalues for the sublevels and generate radial wave function plots, is easy to manipulate once the initial input file is built, and allows students to test a variety of configurations to determine why the configurations provided by the literature are correct; for example, some of the classic problems in general chemistry: how can V form the V^{2+} and V^{5+} cations, is Cu $4s^2$ $3d^9$ or $4s^1$ $3d^{10}$ and why? As OPIUM[26] is a Linux-driven, open-source code, students learning how to use the code inherently become exposed to an operating system which is prevalent in high performance computing,[28] which is a gateway for further development of computational skills. For example, being open-source, the code can be downloaded, examined, modified, and recompiled to include these modifications. As many STEM fields now require some exposure to coding, this permits students to understand how programming can be applied in the field of chemistry.

The original tutorials written for OPIUM are readily available,[26] but are difficult for an introductory level student with minimal exposure to chemistry or coding to follow; they also focus on how to design pseudopotentials, rather than the all-electron calculations. As the all-electron calculations are of particular interest for introducing foundational electronic structure, these tutorials were first updated to enhance readability and are freely available at the Mason Group page, QCC for Everyone.[25] Therefore, for the purposes of supporting the activities described here, modified tutorials were crafted, and later laboratory procedures were designed to help students find answers to class topics they frequently find difficult.

The creation of input files in OPIUM[26] requires knowledge of the *nlm* notation for electron configurations, and the ability to utilize appropriate references to find the atomic radius. The bare necessities of the input file, as shown in the Example Input File section of the student version of the Investigating Atoms and Ions with Computers activity and Figure 2, are a series of three key blocks of information, [Atom], [Pseudo], and [Optinfo], where the latter two deal with functions of the code that are not essential to determining the all-electron energies.

In order to create an input file for carbon, it is necessary to type *emacs C.param* in the command line. This will open up an emacs window, and data can be entered to build the three key blocks required to run calculations. The first block, [Atom], contains the symbol for the element, the total number of sublevels the element possesses, and then the *nlm* notation, electron filling, and a "-" for each sublevel. The "-" is what tells the code to run calculations to determine the orbital energies. Students will enter their information for their test configurations in this key block, which is why they are requested to write and bring their desired test configurations. The second block,

```
[Atom]
C
3
100 2.00 -
200 2.00 -
210 2.00 -

[Pseudo]
3 1.54 1.54 1.54
opt
[Optinfo]
7.07 10
7.07 10
7.07 10
```

Figure 2. Example OPIUM input file for carbon.

[Pseudo] can be used to plot all of an atom's orbital wave functions in Grace.[27] In this block, the number of orbitals one wishes to visualize the plotted wave functions for should be listed, and the cutoff radius to use for each of the wave functions must be included. These orbitals do not all need to be occupied in order to be plotted, they just need to be included in the previous key block with filling values of 0.00 included. For this exercise, using the atomic radius is sufficient to view the wave functions for each orbital within the atom of interest. Traditionally, this block includes just the orbitals one needs to find pseudowave functions and pseudopotentials for as well as the method of calculation. The last block, [Optinfo] contains the technical calculation details, beyond the total energy calculations required for the exercise, which are not relevant to the student. Therefore, we provide sufficient values for this block, without further discussion. Once the input file is edited and saved, a command line prompt is used to run the calculations, which are complete within seconds. The run command can accept a variety of options, and if directed, will evoke Grace to plot the wave functions of the orbitals, providing a way to visualize each orbital's relationship to the atomic nucleus as shown in Figure 3. This command is *./ opium C C.log ae plot wa*

to run the all-electron calculations and plot the all-electron wave functions. The all-electron wave functions differ from the pseudowave functions in that the former exhibit the expected *n-l*-1 number of nodes while the pseudowave functions are smooth and nodeless. The pseudowave functions are not of interest for first-year students, so are not plotted in Figure 3.

Once the graph has been viewed and closed, the log file can be opened using *emacs C.log* in the command line. Once open, one can scroll down to the AE calculation component and obtain the desired data, which includes the energy and the eigenvalues for each orbital. This information is shown in red boxes in Figure 4. All places after the decimal should be recorded as there is a significant difference between Ry/electron and kJ/mol, and a small difference in Rydberg can translate to a significant difference in kJ/mol.

Other test configurations can be ran by copying the input file to a new name using the command *cp C.param C1.param*, and then by using the emacs command to open the new file for modifications. This can be used not only for the various test configurations, but also for testing various cation configurations. Students should write down the names of their files in order to keep their collected data organized.

Determination of Correct Electron Configurations and Radial Wave Functions (Day 1)

When arriving in laboratories, both the introductory and first-year transfer students are assigned to pairs, due to the limited number of laboratory computers and to assist in catching typographical errors. After an initial walk-along demonstration which matches the preactivity video, students are released with directions to take turns to create input files, run calculations, and obtain the necessary data from their output files. Many students use their phones to capture data and graphs obtained from OPIUM[26] and Grace,[27] which allows them to alternate in a timely fashion, as they can record data in their lab notebooks while their partner creates new input files and runs calculations. Upon completing all of the test configurations for their assigned
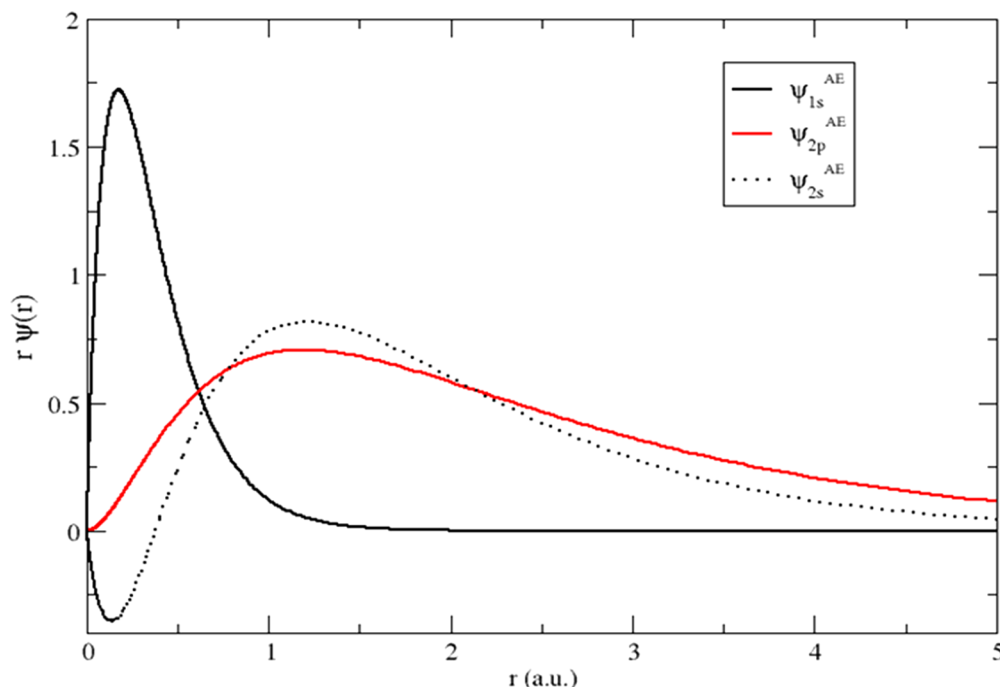

Figure 3. All-electron radial wave functions plotted by Grace for the 1*s*, 2*s*, and 2*p* orbitals of carbon.

```
================================================================
Begin AE calculation
================================================================
Performing non-relativistic AE calculation...
---------Eigenvalue guesses---------
1s          -36.00000000
2s           -2.25000000
2p           -2.25000000

 iter      Etot           Ebs            Ehxc       de_max    dv_max
   1    -78.0101223    -46.1066343    -31.9034880 0.73E+00 0.20E+01
   2    -76.9630569    -44.6770656    -32.2859913 0.10E+00 0.10E+01
   3    -75.3373196    -42.8552064    -32.4821132 0.18E+00 0.14E+00
   .
   .
   .
  25    -74.8485240    -42.5925070    -32.2560171 0.22E-07 0.57E-07
  26    -74.8485240    -42.5925070    -32.2560171 0.11E-07 0.32E-07
  27    -74.8485240    -42.5925069    -32.2560171 0.52E-08 0.18E-07

    27 iterations: (units are Ry and bohr)
Energy:      -74.84852399  Ebs:     -42.59250694  Ehxc:      -32.25601705
Eh   :       35.24955639  Exc:      -9.45350358

Orbital    Filling      Eigenvalues    Norm(rc->oo)     Peak
 |100>      2.000        -19.895705
 |200>      2.000         -1.001950      0.542895      1.224346
 |210>      2.000         -0.398599      0.589443      1.192923


================================================================
End AE calculation
================================================================
```

Figure 4. Image of the all-electron part of the log file for carbon. The relevant information is enclosed in red boxes.

Table 1. Eigenvalue and Total Energy Results for Varying Electron Configurations for Iron

AE:Orbital	Standard Fe Configuration		Alternate Fe Configuration 1		Alternate Fe Configuration 2	
	Filling	Eigenvalues (Ry)	Filling	Eigenvalues (Ry)	Filling	Eigenvalues (Ry)
100	2.00	−509.5785	1.00	−531.9075	2.00	−509.0381
200	2.00	−59.2596	2.00	−64.6049	2.00	−58.6489
210	6.00	−51.1378	6.00	−57.4482	6.00	−50.5343
300	2.00	−6.7576	2.00	−8.2850	2.00	−6.2006
310	6.00	−4.3828	6.00	−5.9158	6.00	−3.8410
320	6.00	−0.5715	6.00	−1.7353	8.00	−0.1214
400	2.00	−0.3823	2.00	−1.0415	0.00	−0.2548
410	0.00	−0.0994	0.00	−0.6400	0.00	−0.0461
Total Energy	−2526.5918 Ry		−2005.8182 Ry		−2526.5907 Ry	

elements, students turn in copies of their data which are entered into an Excel file to create a class database. The students then compare the configurations for each element and are asked to determine the correct configuration without referring to their textbook or the Internet, and also to explain how they can identify it is correct if they did not have the correct electron configuration written down in front of them. If students struggle with this, the discussion is opened up to the class, and the greater pool of students is able to recognize the configuration possessing the lowest total energy can be identified as the correct one. Some sample student test configurations are shown in Table 1 for Iron. In Table 1, there are three sets of student data which were

collected for their test configurations. The first of these configurations is the correct one, the configuration in the middle is a student test configuration where a mistype or mistake occurred in entering the filling, or in writing the electron configuration, and in the last configuration, the student moved some electrons from the 4s orbital into the 3d orbital. A comparison of the total energies should communicate to the student that a mistake was made in Alternate Fe Configuration 1 as a significantly different energy is observed. By comparing the total energies of the Standard Fe Configuration and Alternate Fe Configuration 2, the first can be identified as being correct as it has the lowest overall energy out of the test configurations.

From the class database, the students extract the energies for the 4s and 3d sublevels as shown in Table 2. As students are not

Table 2. 3d and 4s Eigenvalues for the 3d Transition Metals

Element	3d Eigenvalues (Ry)	4s Eigenvalues (Ry)
Sc	−0.2619	−0.3135
Ti	−0.3397	−0.3348
V	−0.4089	−0.3525
Cr	−0.3016	−0.2363
Mn	−0.5327	−0.3828
Fe	−0.5897	−0.3965
Co	−0.6215	−0.4095
Ni	−0.6969	−0.4220
Cu	−0.4044	−0.3447
Zn	−0.7973	−0.4459

always comfortable analyzing data from a table, they are required to create a visual representation as seen in Figure 5, and then asked to explain what information they can derive from both the graph and table about electron removal and the Aufbau principle. There will be differences in the eigenvalues depending on whether or not the students include the 4p sublevels in their calculations.

Students are also asked to examine their collected graphs to identify how many radial nodes are contained by each of the s orbitals, and to determine an equation on how this can be related to the principle quantum number. An example of a graph they used for a 3d transition metal is shown in Figure 6. Once the students have changed the colors to allow them to identify each wave function, they can take notes on how many times each wave function crosses zero. When this information is recorded, the students can organize their data and write out what they think the relationships are between energy levels, sublevels, and the number of nodes they should possess.

Determination Ionization Energies (Day 2)

For students enrolled in the first-year transfer course, not only is a greater understanding of the Aufbau principle and orbital filling desired, but also the ability to calculate ionization energies

for cations and analyze the error in these calculations to determine what is considered acceptable for use in other calculations such as lattice energies. Therefore, the transfer course picks up with the modification of their input files from the first week to create input files for an array of cations, also allowing them to look at the correct electron configurations for the different cations. This required the students to complete calculations and extract the total energies for the neutral atom and related cations. The difference between the total energies can be found; for example, for Na, the energy of the neutral atom is subtracted from the energy of the Na^+ cation as shown below:

$$-322.486965 \ Ry - -322.866737 \ Ry = 0.3797722 \ Ry \tag{1}$$

To determine the percent errors for the ionization energies, students are required to convert the Ry/atom difference in energy found in eq 1 to kJ/mol (1 Ry = 1312.75 kJ/mol) as shown in eq 2, and to use reliable sources, such as the NIST Chemistry Webbook,[29] to find the literature values. Here we show an example unit conversion in eq 2 followed by the percent error calculation in eq 3 for the determination of sodium's ionization energy.

$$0.3797722 \ Ry \times \frac{1312.75 \ \frac{kJ}{mol}}{Ry} = 498.54 \ \frac{kJ}{mol} \tag{2}$$

$$\frac{\left| 496 \ \frac{kJ}{mol} - 498.54 \ \frac{kJ}{mol} \right|}{496 \ \frac{kJ}{mol}} \times 100 = 0.512\% \tag{3}$$

By having the students find literature values for their percent error calculations, it opens up discussion of what resources can be considered reliable, and which ones are not acceptable and why they are not considered to be. Students are then asked to explain why the order in which electrons should be removed is important to obtaining accurate ionization energies. Data is again submitted to a class data file for use in future lectures and assignments discussing the hybridization of orbitals, bonding, crystal lattice formation, and in the calculations of lattice
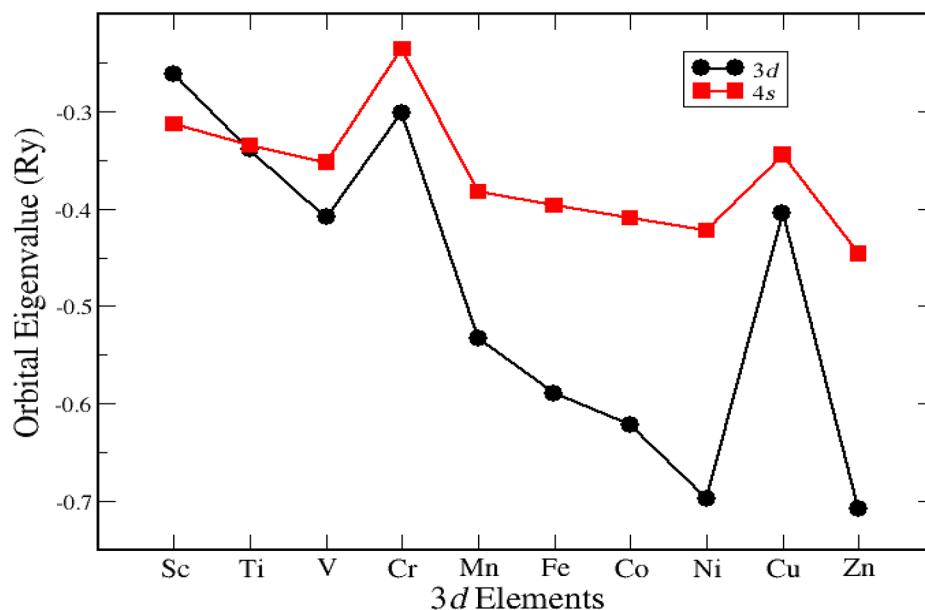

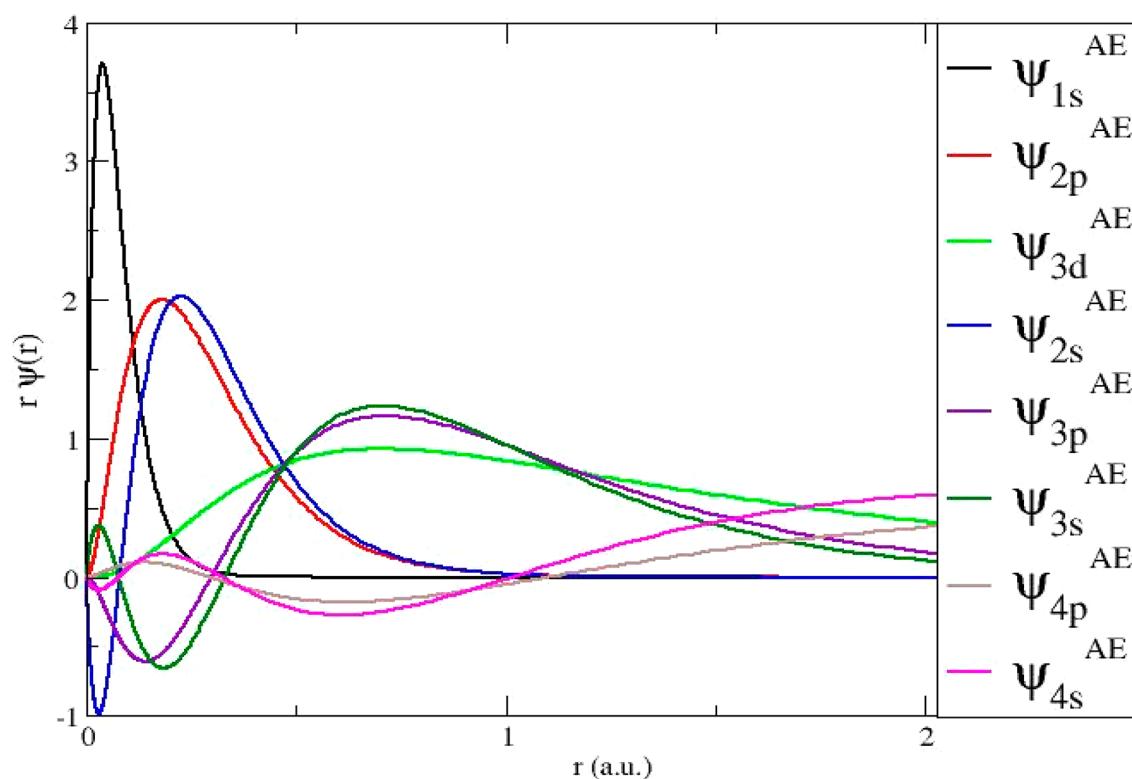**Figure 5.** Graph comparing the eigenvalues of the valence orbitals for the 3d transition metals.

Figure 6. Graph of the all-electron radial wave functions for a transition metal.

energies. If students happen to be assigned the same elements, such as the case when dealing with larger classes, they are encouraged to compare their data as they should be obtaining identical results if using the same test configurations.

HAZARDS

There are no procedural or chemical hazards for this laboratory activity.

DISCUSSION

The activity provided students a means of learning the abstract topics of the Aufbau principle, electron configurations, and related introductory quantum mechanics through an interactive approach. Students learned how to use command line prompts to run the open-source software, OPIUM[26] and Grace,[27] to obtain data relating to fundamental quantum mechanics. From the collected data, students were able to infer which electron configurations can be considered to be correct based off the computed total energies, collected and analyzed data to understand the differences between energy levels and sublevels, and created graphs to help them explain observed trends. Additionally, this activity provides students with a practical application of computational chemistry that does not require an advanced understanding of quantum mechanics, while providing them with valuable workplace skills.

Student Feedback and Engagement

As this lab activity was tested with more than 300 introductory-level students over the course of 5 semesters (Spring 2018–Spring 2020), there was ample opportunity for them to provide feedback. The majority of the students had no exposure to using command line prompts, using Linux, or any sort of computational chemistry resources beyond online homework systems. As the activity was designed and tested, the students suggested

some modifications which were implemented, including a video on how to use the software, rather than just background reading material. The lab activity was met with a mix of enthusiasm and frustration. For the students planning to go into fields requiring the use of data analysis tools, they found it interesting that software could be used to obtain data that was previously unavailable to them and that it could be used to help explain abstract topics such as quantum mechanics. However, those who were less technologically savvy found the exercise frustrating as the software is case-sensitive and typographical mistakes lead to numerous errors. Overall, when students were asked to reflect on how they thought the activity enhanced their understanding of the Aufbau principle and electron configurations, the majority of the students (75%) from the Fall 2019 semester answered yes. The students were required to include graphs and answers to their data analysis questions (Supporting Information) in their lab reports. Their reports provided a formative assessment of student learning. Additionally, multiple configurations for elements have been included as short answer questions on exams, as shown in Table 1. Questions relating to the sample data shown in Table 1 required students identify the correct electron configuration and explain why that particular configuration as the correct one as well as sketch out an electron filling diagram. Students in the introductory class had the most trouble with creating graphs of their data (45%), as the large differences in energy can result in data points appearing very close together. Additionally, these students frequently struggled with magnitude as they tended to ignore or drop the negative sign, which made it challenging for them to determine the lowest energy configuration. Through examining data from varying configurations such as observed in Table 1, and class discussions on magnitude, students were able to determine which configuration possesses the lowest total energy, making it the most stable configuration. After collecting and analyzing data

from not only their assigned elements, but also the class database, 55% of the students included graphs in their reports which showed the d orbitals being lower in energy than the s orbitals. The data used to obtain this graph are found in Table 1, and Figure 5 is a replica of the graphs the students created in their lab reports. As many students struggled with creating these graphs, Figure 5 was discussed in class to help the students who struggled with graph creation and to ensure students could come to a conclusion on why the removal of the s electrons versus d electrons is likely to occur.

Additionally, students gained experience with creating and analyzing graphs, and were able to calculate total energies for the anomalies in the Aufbau principle such as molybdenum ($5s^2\,4d^4$ vs $5s^1\,4d^5$) and silver ($5s^2\,4d^9$ vs $5s^1\,4d^{10}$) and similar elements through the use of data rather than relying on rote-memorization. The Grace plots of the wave functions, as shown in Figure 6, provided students with a better visualization of how orbitals can be packed around the nucleus as the principle quantum number increases.

For the students who also determined the ionization energy, they learned how the software can be used to calculate first, second, and higher order ionization energies which can be difficult to find in first-year texts, and compared them to accepted literature values. Based off of the calculated percent errors using known configurations and literature values, students were requested to decide whether or not their ionization energies were acceptable for use in calculations, and how computational chemistry can be used to derive missing information. This information can then be used in future calculations in first-year chemistry if desired, such as in the determination of lattice energies. Additionally, the data and trends observed can be used as a basis for teaching molecular orbitals and transition metal chemistry.

CONCLUSIONS

In summary, we presented a laboratory activity that provided students with a hands-on approach to examining the Aufbau principle through the use of open-source software to students in first-year courses. Thanks to the advances in technology, the mathematical background required to solve Schrödinger's equation by hand is no longer a barrier, allowing students in introductory level courses to explore these topics and develop their own explanation for the Aufbau principle and electron configurations, rather than relying on rote memorization. Through this activity, students were able to use open-source software to seek answers to their questions regarding electron configurations, using an interactive approach to try to identify and explain anomalies in their configurations through data collection and analysis. Students learned how to obtain wave function plots and how to identify nodes and suggest an equation for how nodes are related to principle quantum numbers through using their data rather than relying on rote memorization. Through data analysis and graphical comparison, students learn to create a visual aid to determine the order of electron removal and why the $4s$ electrons are removed prior to the $3d$ electrons. Students used the software to seek data to build orbital energy diagrams rather than relying on rote memorization. Additionally, this activity provided students with valuable skills for the workforce through the use of data extraction and interpretation. By having the students perform calculations exploring abstract concepts of quantum numbers and electron configurations, educators were able to tackle one of the educational challenges of first-year chemistry courses while

engaging quantitative reasoning and critical thinking skills rather than pure memorization. The inclusion of computational chemistry though the use of open-source codes not only provides students a hands-on approach to understanding difficult concepts, but can assist in creating a well-rounded introductory chemistry curriculum. In regards to the learning process, student reflections in their lab report show the majority of the students believe this exercise is enhancing their understanding of the anomalies in electron configurations and why they are observed. Students demonstrated an ability to determine correct electron configurations from obtained data and thought themselves to have a stronger understanding in electron removal trends. The building of critical thinking and quantitative reasoning skills were also encouraged through the facilitation of discussions and by permitting students the opportunity to revise reports afterward if desired. When used at the community college, this activity helps create a strong foundation for students who plan to transfer to institutions in which computing plays a larger role in the curriculum, specifically for students who will later move on to advanced courses such as physical and inorganic chemistry.

ASSOCIATED CONTENT

Ⓢ Supporting Information

The Supporting Information is available at https://pubs.acs.org/doi/10.1021/acs.jchemed.1c00700.

> Instructor materials (PDF) (DOCX)
> Student materials (PDF) (DOCX)
> OPIUM calculation overview (quiz question answers are part of the file, and should be removed from the document prior to uploading it for student access) (PDF)

AUTHOR INFORMATION

Corresponding Author

Sara E. Mason — *Department of Chemistry, University of Iowa, Iowa City, Iowa 52242, United States*; Email: sara-mason@uiowa.edu

Authors

Irene K. Metz — *Department of Chemistry, University of Iowa, Iowa City, Iowa 52242, United States; School of Science and Health Sciences, Hawkeye Community College, Waterloo, Iowa 50701, United States*; ⓞ orcid.org/0000-0001-9001-2395

Joseph W. Bennett — *Department of Chemistry, University of Iowa, Iowa City, Iowa 52242, United States*; ⓞ orcid.org/0000-0002-7971-4772

Complete contact information is available at:
https://pubs.acs.org/10.1021/acs.jchemed.1c00700

Notes

The authors declare no competing financial interest.

■ REFERENCES

(1) Williams, D. P.; Woodward, J. R.; Symons, S. L.; Davies, D. L. A Tiny Adventure: The introduction of problem based learning in an undergraduate chemistry course. *Chem. Educ. Res. Pract.* **2010**, *11* (1), 33−42.

(2) Partanen, L. How student-centred teaching in quantum chemistry affects students' experiences of learning and motivation—a self-determination theory perspective. *Chem. Educ. Res. Pract.* **2020**, *21* (1), 79−94.

(3) Partanen, L. Student-centred active learning approaches to teaching quantum chemistry and spectroscopy: quantitative results from a two-year action research study. *Chem. Educ. Res. Pract.* **2018**, *19*, 885−904.

(4) Feller, S. E.; Dallinger, R. F.; Caylor McKinney, P. A Program of Computational Chemistry Exercises for the First-Semester General Chemistry Course. *J. Chem. Educ.* **2004**, *81* (2), 283−287.

(5) Pearson, J. K. Introducing the Practical Aspects of Computational Chemistry to Undergraduate Chemistry Students. *J. Chem. Educ.* **2007**, *84* (8), 1323−1325.

(6) Esselman, B. J.; Block, S. B. VSEPR-Plus: Correct Molecular and Electronic Structures Can Lead to Better Student Conceptual Models. *J. Chem. Educ.* **2019**, *96* (1), 75−81.

(7) Grushow, A.; Reeves, M. S. Using Computational Methods To Teach Chemical Principles: Overview. In *Using Computational Methods To Teach Chemical Principles*; American Chemical Society: 2019; Vol. *1312*, pp 1−10.

(8) Hanwell, M. D.; Curtis, D. E.; Lonie, D. C.; Vandermeersch, T.; Zurek, E.; Hutchison, G. R. Avogadro: An advanced semantic chemical editor, visualization, and analysis platform. *J. Cheminf.* **2012**, DOI: 10.1186/1758-2946-4-17.

(9) Valiev, M.; Bylaska, E. J.; Govind, N.; Kowalski, K.; Straatsma, T. P.; Van Dam, H. J. J.; Wang, D.; Nieplocha, J.; Apra, E.; Windus, T. L.; De Jong, W. A. NWChem: A comprehensive and scalable open-source solution for large scale molecular simulations. *Comput. Phys. Commun.* **2010**, *181* (9), 1477−1489.

(10) *Open Chemistry*. https://www.openchemistry.org/ (accessed April 2021).

(11) *Jmol: An Open-Source Java Viewer for Chemical Structures in 3D*. http://jmol.sourceforge.net/ (accessed April 2021).

(12) Mohebifar, M.; Sajadi, F. Chemozart: A web-based 3D molecular structure editor and visualizer platform. *J. Cheminf.* **2015**, *7* (1), 56.

(13) Esselman, B. J.; Hill, N. J. Integration of Computational Chemistry into the Undergraduate Organic Chemistry Laboratory Curriculum. *J. Chem. Educ.* **2016**, *93* (5), 932−936.

(14) Ruddick, K. R.; Parrill, A. L.; Petersen, R. L. Introductory Molecular Orbital Theory: An Honors General Chemistry Computational Lab as Implemented Using Three-Dimensional Modeling Software. *J. Chem. Educ.* **2012**, *89*, 1358−1363.

(15) Ge, Y. Let Students Derive, by Themselves, Two-Dimensional Atomic and Molecular Quantum Chemistry from Scratch. *J. Chem. Educ.* **2016**, *93* (12), 2033−2039.

(16) Natoli, S. N.; McMillin, D. R. Demonstrating Hund's Rule in Action by Exploring the Magnetic Properties of Metal Complexes with 3dn and 4fn Configurations. *J. Chem. Educ.* **2018**, *95* (1), 126−130.

(17) Hoffman, G. G. Using an Advanced Computational Laboratory Experiment To Extend and Deepen Physical Chemistry Students' Understanding of Atomic Structure. *J. Chem. Educ.* **2015**, *92* (6), 1076−1080.

(18) Le, A.; Marquino, G. J.; Ge, Y. An Inquiry-Based Molecular Modeling Laboratory for Students to Discover VSEPR Theory and Resonance Structures. *Chem. Educ.* **2019**, *24*, 59−61.

(19) Carroll, B.; Lehrman, A. The electron configuration of the ground state of the elements. *J. Chem. Educ.* **1948**, *25* (12), 662−666.

(20) Mayer, M. G. Rare-earth and transuranic elements. *Phys. Rev.* **1941**, *60* (3), 184−187.

(21) Meggers, W. F. Electron configurations of ″rare-earth″ elements. *Science* **1947**, *105* (2733), 514−516.

(22) Vanquickenborne, L. G.; Pierloot, K.; Devoghel, D. Transition Metals and the Aufbau Principle. *J. Chem. Educ.* **1994**, *71* (6), 469−471.

(23) Pearson, W. H. A General Chemistry Laboratory Experiment Relating Electron Configuration and Magnetic Behavior. *J. Chem. Educ.* **2014**, *91* (1), 116−118.

(24) Elkins, K. M. Teaching Quantum Numbers Using Candy: An Activity for a General Chemistry Course. *Chem. Educ.* **2014**, *19*, 229−231.

(25) *QCC for Everyone*; Mason Group. https://masongroup.lab.uiowa.edu/qcc-everyone (accessed June 2019).

(26) *Opium: Pseudopotential Generation Project*. http://opium.sourceforge.net/ (accessed June 2018).

(27) *Grace*. https://plasma-gate.weizmann.ac.il/Grace/ (accessed June 2018).

(28) Towns, J.; Cockerill, T.; Dahan, M.; Foster, I.; Gaither, K.; Grimshaw, A.; Hazlewood, V.; Lathrop, S.; Lifka, D.; Peterson, G. D.; Roskies, R.; Scott, J. R.; Wilkins-Diehr, N. XSEDE: Accelerating Scientific Discovery. *Comput. Sci. Eng.* **2014**, *16* (5), 62−74.

(29) *NIST Chemistry WebBook*. https://webbook.nist.gov/chemistry/ (accessed April 2018).